Planet Scale Software Updates

Milan Vojnović

Microsoft Research

Collaborators: Christos Gkantsidis, Thomas Karagiannis, Pablo Rodriguez

Acknowledgment

- Thanks to
  - Ryan Adul
  - Josh Dunn
  - Deighton Maragh
  - Kurt Parent
  - Mark Roellich
  - Rob Satterwhite

  for answering our numerous questions and providing us with data

Outline

- Overview of WU service and data
- Traffic characterisation
  - Update properties
  - User properties
- Dissemination strategies
  - Caching
  - Peer-to-peer w/o locality
- Summary

Summary of Findings

- User/Traffic Characteristics
  - Always-on-machines ≈ 20%
  - > 90% users use automatic updates
  - ≈ 80% of IPs check for updates within a day
  - User query arrival are bursty
- Patch characteristics
  - Many patches (ex SP2, ≈ 2000)
  - But a small number of patch groups (order of magnitude smaller than number of patches)
  - A few groups cover most users (a few distribution channels)

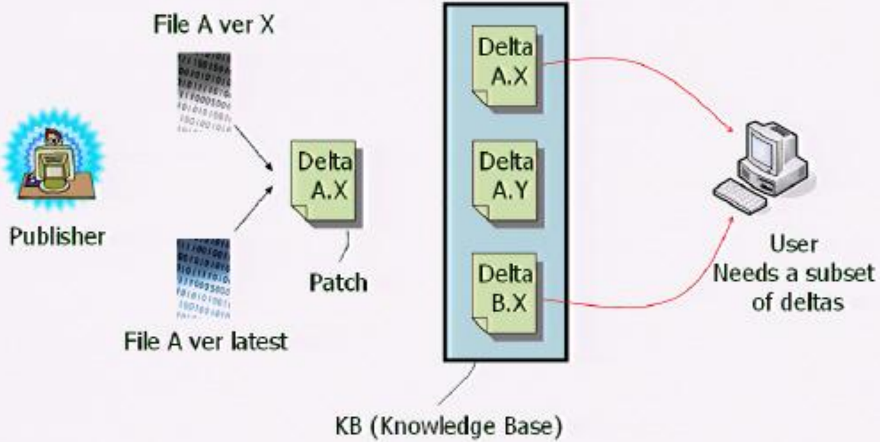
Summary of Findings (cont'd)

- Server workload reduction with caching
  - ≈ 25-35% of users (estimate)
- P2P
  - Efficient reduction of central server load
  - Efficient serving of flashcrowd demand
  - Increase of ISP upload traffic
  - But this is effectively mitigated by P2P with locality
    - Characterisation of the reduction
    - Key factors: user query arrival rate, user online time, distribution of users over ISPs

Outline

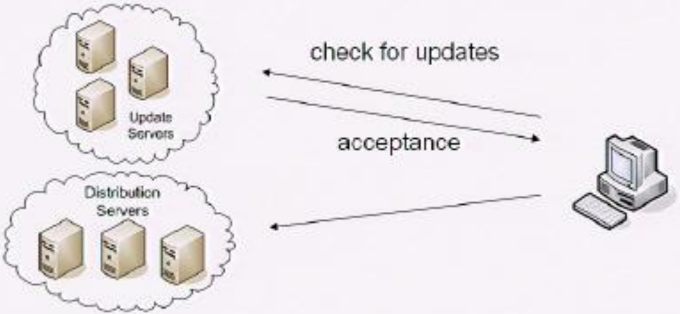
- Overview of WU service and data
- Traffic characterisation
  - Update properties
  - User properties
- Dissemination strategies
  - Caching
  - Peer-to-peer w/o locality
- Summary

Windows Update: Overview



Windows Update: Overview (cont'd)

- Users PULL patches from central server



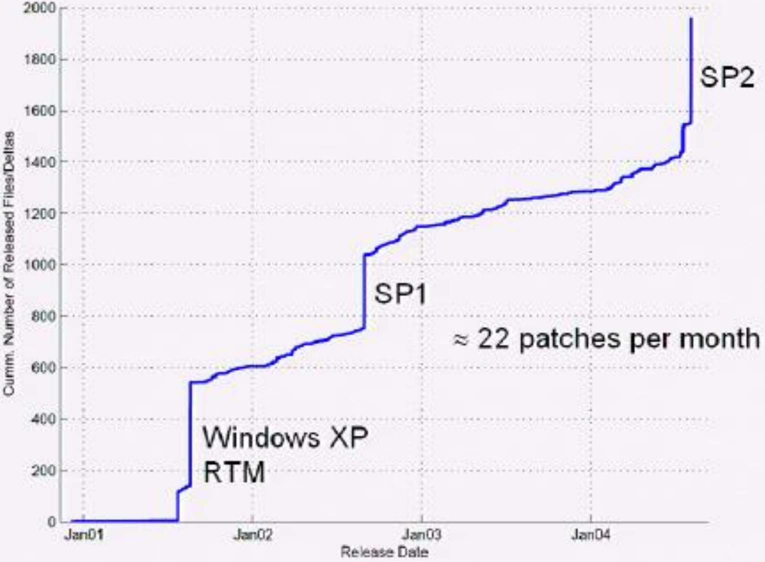
Data Used in Our Analysis

- Distribution server
  - Service Pack (SP)
    - IIS logs from one server (2 days shortly after SP2 release)
    - Packet level traces from one server (2 days, 6 months after SP2 release)
    - Source code info (e.g. modified files, deltas)
  - Tuesday Patch (TP)
    - IIS logs from one server
    - Jun 05 (1 day), Aug 05 (1 day), Oct 05 (14 days)
    - One day ≈ 300,000 IPs
- Update server
  - IIS logs from ALL update servers (Jan 4-6, 06)
  - 300M users per day
  - 150 update servers
  - 2B requests
- Other
  - XP release build tree

Outline

- Overview of WU service and data
- Traffic characterisation
  - Update properties
  - User properties
- Dissemination strategies
  - Caching
  - Peer-to-peer w/o locality
- Summary

Patch Release Times



Groups of Patches

- Goal: Identify groups of patches
  - Helps reducing publishing complexity
  - Ex reduction of multicast channels
- Method:
  - $x_i(u) = 1$ if user u downloads patch **i**, else $x_i(u) = 0$
  - For each pair of patches i,j compute the cosine correlation factor:

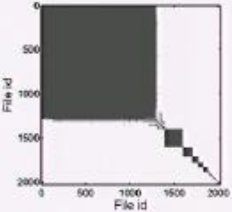
$$c_{ij} = \frac{x_i^T x_j}{\sqrt{x_i^T x_i}\sqrt{x_j^T x_j}}$$

  - Threshold rule: if $c_{ij} > 0.9$ then patches i and j regarded "related"

Groups of Patches (cont'd)

- Data: SP2
  - 2129 users (conditional on that user requested > 1000 files)
  - 2029 files
- Graph (V,E)
  - V = set of patches
  - E = edges
  - (i,j) is an edge iff $c_{ij} > 0.9$
- Group = connected component in graph G

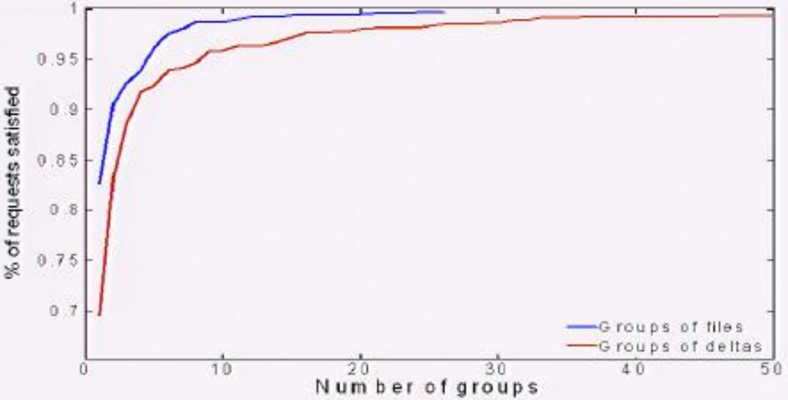
Adjacency matrix:



- Result:
  - 26 (disjoint) groups cover 2003 files
  - Largest group = 1460 files
- Natural interpretation
  - OS versions, ex core, Pro, Home
  - OS components, ex WMP, IIS, etc

Benefits of Patch Grouping

- Few groups cover most user requests

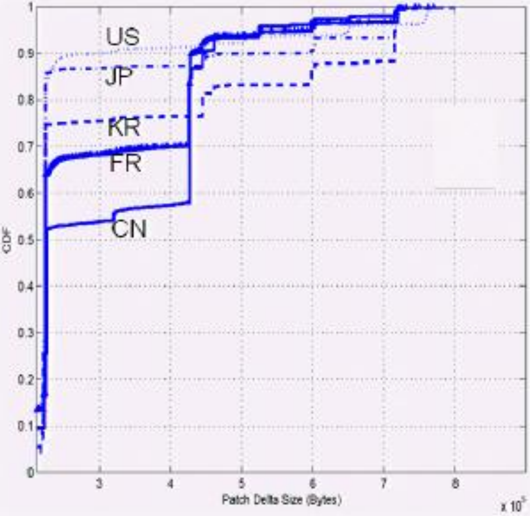


Outline

- Overview of WU service and data
- Traffic characterisation
  - Update properties
  - User properties
- Dissemination strategies
  - Caching
  - Peer-to-peer w/o locality
- Summary

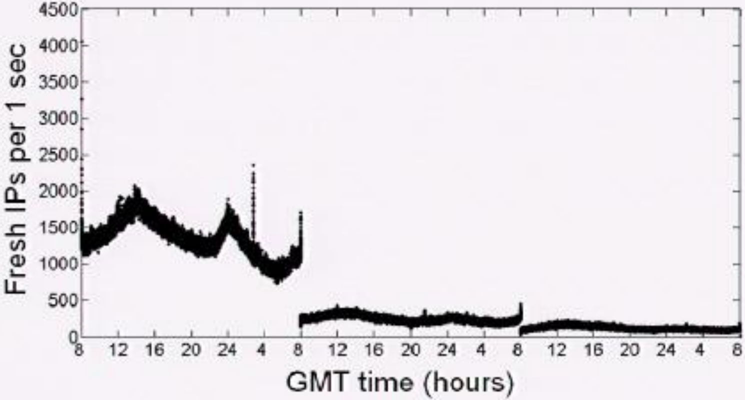
How Much Up-to-date User Computers Are ?

- Metric: patch size
  - The smaller the delta size, the smaller the difference with the latest file
  - Validated
- Data: TP Jun 05
  - KB896358
- Observations:
  - A few small deltas cover large fraction of users
  - Some geographical diversity



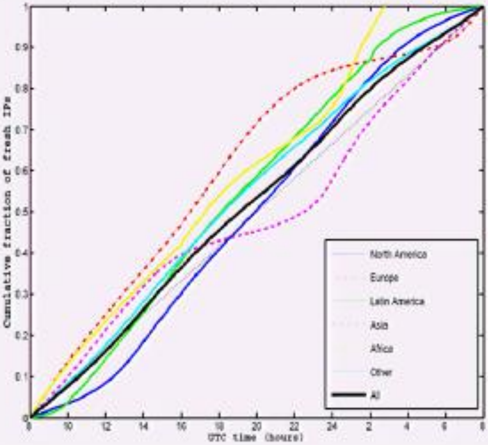
Fresh IPs

- Most fresh IPs arrive in day 1
- 117M in day 1, 22M in day 2, 11M in day 3
- Peaks are due to timezone effects



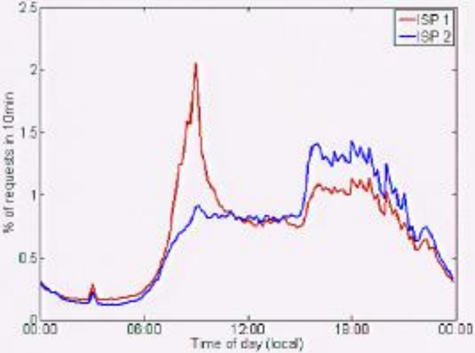
User Query Arrivals Over Time

- Continent aggregates
- Time-of-day effect



Burstiness of User Queries

- Burstiness expected: human activity
  - Query initiated shortly after computer boot time
- Evidence
  - User query arrivals for two geographically-collocated ISPs

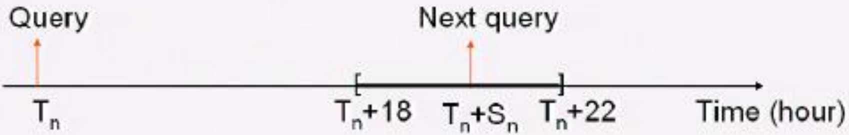


Burstiness of User Queries (cont'd)

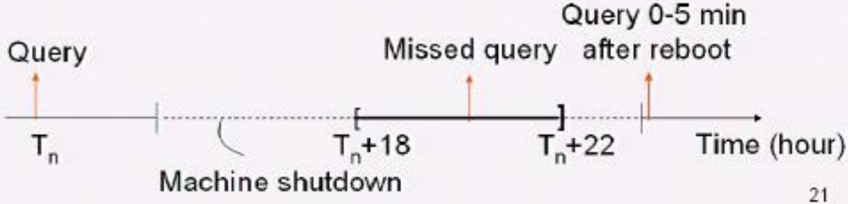
- Test-hypothesis:
  - (H_0) Query arrival rates of AS aggregates is uniform in time
- Data: update-server logs
- Kolmogorov-Smirnov test result:
  - H_0 cannot be rejected for 52% ASs
  - But these ASs cover less than 0.1% of users
- Suggests: AS aggregates of queries are non uniform

Types of User Queries

- Always-on-machine (AOM)

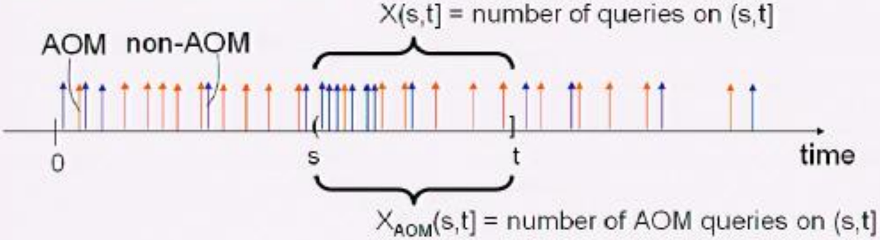


- Non AOM



Problem: How Many Machines are AOM ?

Number of queries on (s,t]

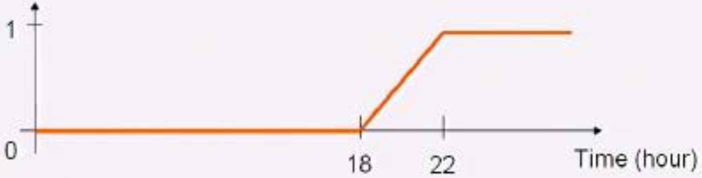


- Can observe X(s,t]
- Can't observe X_{AOM}(s,t]

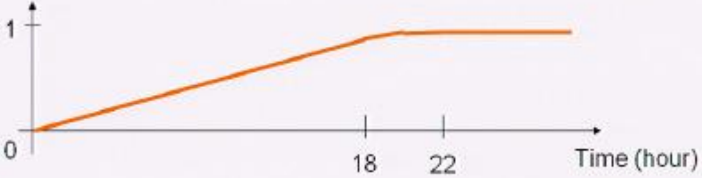
- Problem: Estimate the unknown number of AOM machines

AOM Facts

- Inter-query time S_n ~ Uniform(18,22)



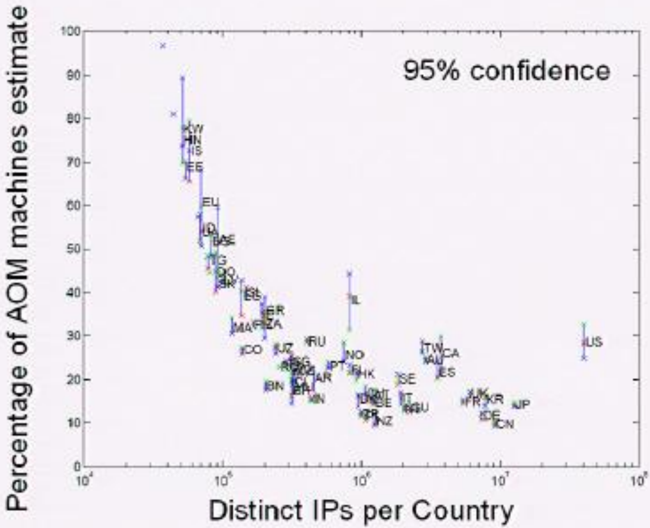
- Query residual time distribution
  - (time until next user query observed from an arbitrary time)



Estimation of the Number of AOM Hosts (cont'd)

- Unknown: M (number of AOM machines)
- Known:
  - A(t) (host query residual time CDF)
  - $X_{AOM}(s,t] \sim$ Binomial (M, A(t)-A(s))
- Observation: X(s,t]
- Estimator: $\hat{M} = \min_{t \in T} \frac{X(t_k, t_{k+1}]}{A(t_{k+1}) - A(t_k)}$
  - T := finite set of distinct times
- Leverages the facts:
  - $X_{AOM}(s,t]$ concentrates around M(A(t)-A(s)), for large M
  - non-AOM overnight activity low
- Not on slides: estimator for a fixed confidence

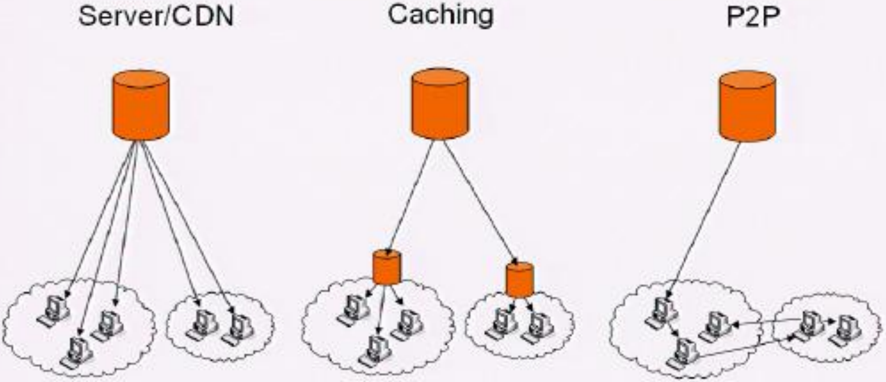
AOM Machines per Country



Outline

- Overview of WU service and data
- Traffic characterisation
  - Update properties
  - User properties
- Dissemination strategies
  - Caching
  - Peer-to-peer w/o locality
- Summary

Dissemination Strategies



Current approach

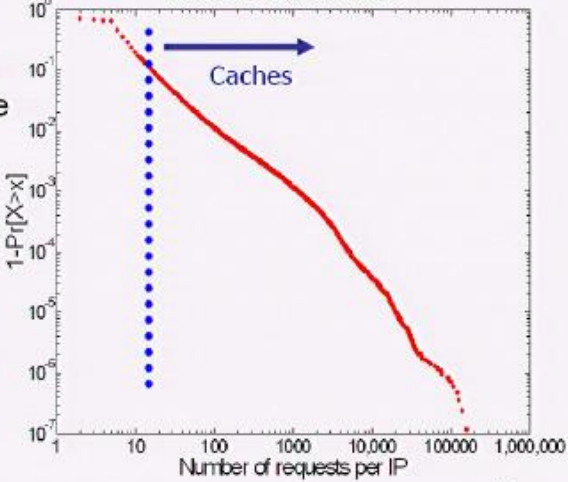
Not currently supported (byte range requests)

Future ?

Inference of Caches

- We cannot observe whether a query is from a user machine or cache
- Inference
  - Threshold k
  - If the number of requests from an IP > k, then classify this IP as a cache

Cache size CCDF



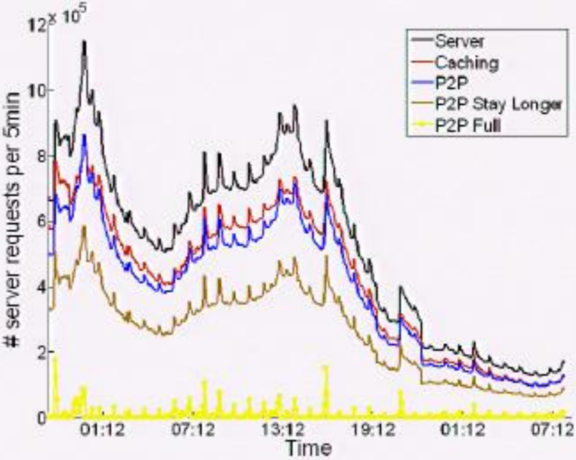
Benefits of Caching

- Data: update-server logs

Threshold	# Caches	Users covered	% Users covered
> 2	≈ 25M	≈ 187M	62%
> 5	≈ 5.5M	≈ 116M	38%
> 10	≈ 1.7M	≈ 88.5M	29%
> 50	≈ 210K	≈ 60M	20%

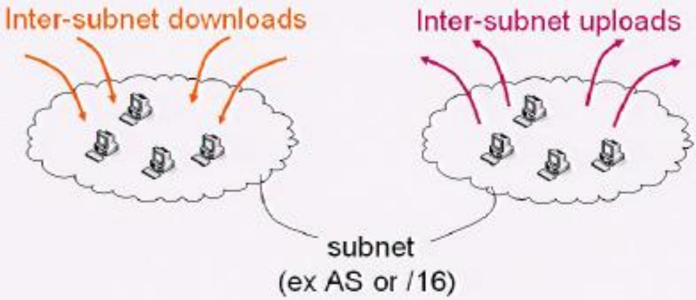
Server Workload

- Data: Jan 06 TP
- Caching inferred with threshold > 25
- User downlink/uplink capacity ratio = 4
- Server covers residual downlink capacity
- Performance with P2P depends on how long users stay online
- Future work: dimensioning rules for service provisioning



Inter-Subnet Traffic

- Of interest from network perspective
  - ex ISP

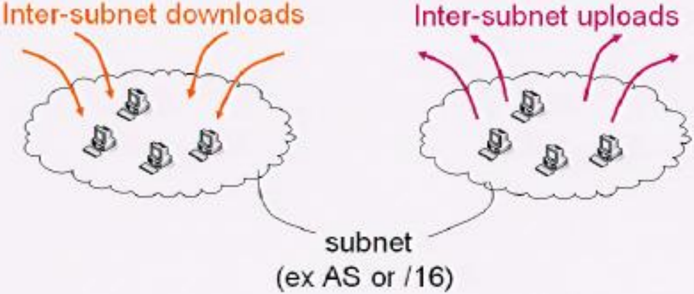


Inter-Subnet Traffic (cont'd)

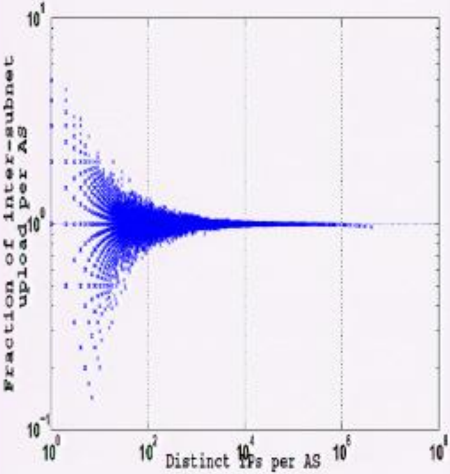
- Server/CDN
  - Proportional to the number of hosts
- Caching
  - Only the first request for an item from a subnet directed to server
  - Further requests for this item directed to the cache in local subnet
- P2P
  - External downloads to a subnet proportional to number of hosts in local subnet
  - Effectively the same external download as with Server/CDN
  - Plus increase of external uploads!

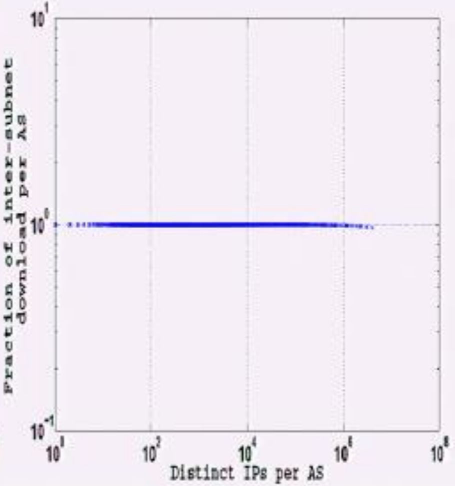
Inter-Subnet Traffic

- Of interest from network perspective
  - ex ISP



P2P-Induced Per-Subnet Upload and Download Traffic





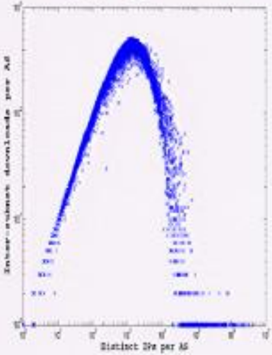
Reduction of Inter-Subnet Traffic: P2P with Locality

- As P2P plus local preference
  - Host download request directed to an online local host, whenever there exists one
  - Else, directed to a random online host
- Effective reduction of server load
- Reduction of inter-subnet download & upload

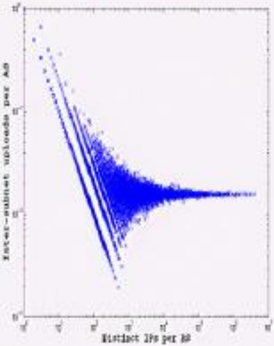
- What are the system key factors that determine the inter-subnet traffic reduction ?
- Can I estimate the reduction for given factors that characterise the system ?

P2P with Locality

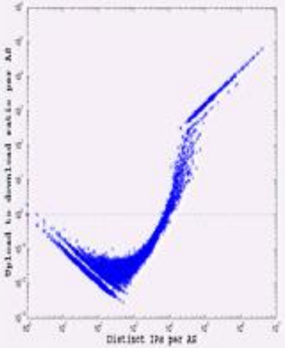
- Subnet = AS domain
- Each host remains online for 1 min
- Mapping hosts to subnets from IIS update-server logs



Per-subnet external download



Per-subnet external upload



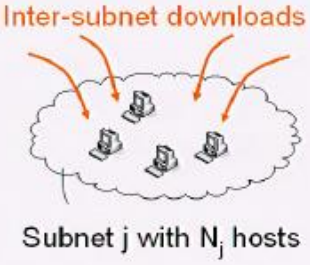
Per-subnet upload to download asymmetry

P2P with Locality: Per-subnet Download

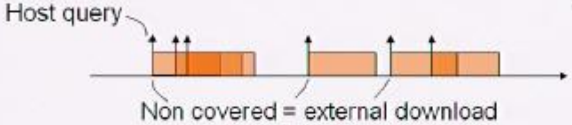
- A host queries at random time with density a
- A host remains online for random time with distribution B

- Expected inter-subnet downloads:

$$\mathbf{E}(D_j(t)) = N_j \int_0^t (1 - \bar{B} * a(s))^{N_j - 1} a(s) ds$$

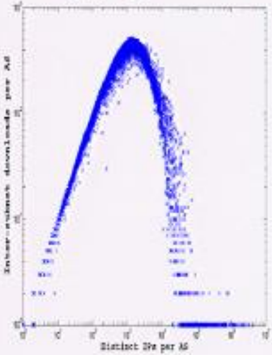


- Cover problem:

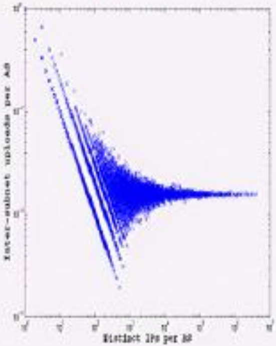


P2P with Locality

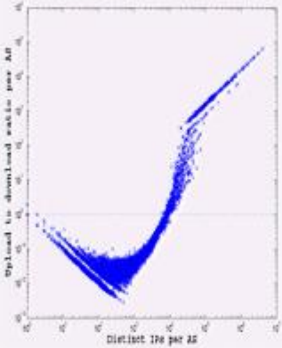
- Subnet = AS domain
- Each host remains online for 1 min
- Mapping hosts to subnets from IIS update-server logs



Per-subnet external download



Per-subnet external upload



Per-subnet upload to download asymmetry

P2P with Locality: Per-subnet Upload

- Subnet size distribution:

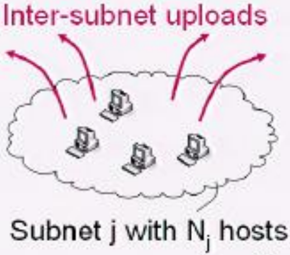
$$\nu(i) := \frac{\text{number of subnets with } i \text{ hosts} * i}{\text{number of hosts}}$$

- Expected inter-subnet uploads:

$$\mathbf{E}(U_j(t)) = N_j R_t$$

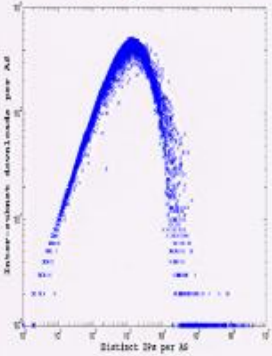
$$R_t = \int_0^t \mathbf{E}((1-\overline{B} * a(s))^{S-1})a(s)ds$$

$$S \sim \nu$$

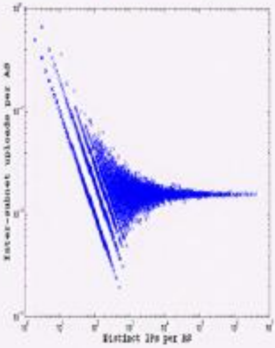


P2P with Locality

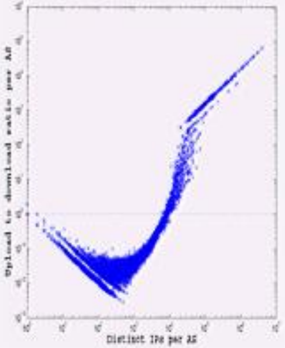
- Subnet = AS domain
- Each host remains online for 1 min
- Mapping hosts to subnets from IIS update-server logs



Per-subnet external download

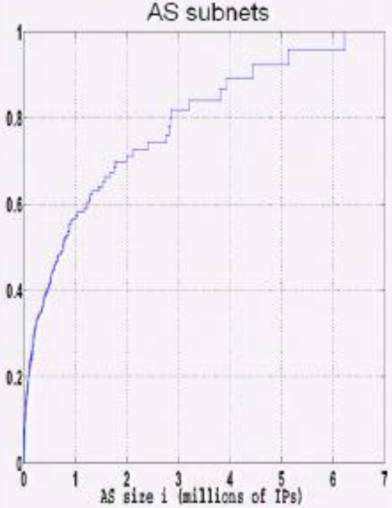


Per-subnet external upload

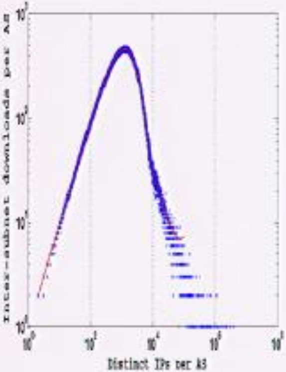


Per-subnet upload to download asymmetry

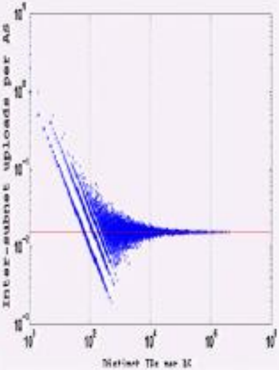
Key Factor: Subnet size distribution ν (cont'd)



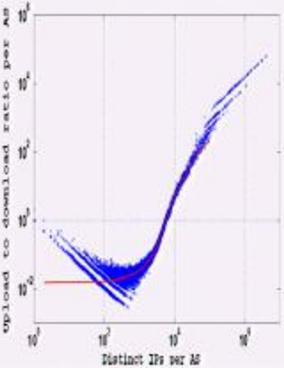
P2P with Locality: Analysis vs Experiments



Per-subnet external download



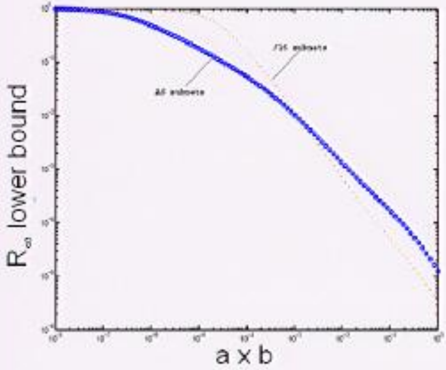
Per-subnet external upload



Per-subnet upload to download asymmetry

P2P with Locality: Inter-Subnet Traffic Reduction

- AOM hosts
- Host query rate a = 1/20 per hour
- Host online time b = 1 min
- R_∞ = upload reduction factor



Reduction	AS	/16
0.1	2.15 sec	14 sec
0.01	72 sec	72 sec
0.001	24 min	10 min
0.0001	4 hours	72 min

Outline

- Overview of WU service and data
- Traffic characterisation
  - Update properties
  - User properties
- Dissemination strategies
  - Caching
  - Peer-to-peer w/o locality
- Summary

Summary

- Software updates critical and popular
  - ≈ 300M queries per day
  - ≈ 90% users use automatic updates
- Grouping of patches
  - A few groups cover most users
  - Reduces distribution complexity
- Users characterisation
  - Most come within one day
  - Most are up-to-date
  - ≈ 20% always-on-machines
- Traffic characterisation of user queries
  - Bursty (time of day effects)